2011 Seventh International Conference on Computational Intelligence and Security

Hybrid Genetic Algorithm for TSP

Lai Nian
Institute of Information Engineering
Xiangtan University
Xiangtan, China 411105
E-mail: 250572270@qq.com

Zheng Jinhua
Institute of Information Engineering
Xiangtan University
Xiangtan, China 411105
E-mail: jhzheng@xtu.edu.cn

***Abstract*--When use simple genetic algorithm for solving the traveling salesman problem, the generated optimal solution is over stochastic and does not consider the neighborhood information in whole search process. In order to reduce the randomness, the paper proposes a hybrid genetic algorithm which based on ant algorithm that making better use of the inspiration information of previous generations. In addition, it adds a local search process so that more useful information is supplied to get the optimal solution.**

Keywords-traveling salesman problem; hybrid genetic algorithm; ant algorithm; local search

I. INTRODUCTION

Traveling salesman problem (TSP) [1] is a well-known classic problem in the optimization problem, yet has not been completely resolved, has been classified as NP-complete problems. Its general formulation is that there are a number of businessmen who are heading the city selling goods, starting from the city 1, getting to the rest of the city once and only once, then back to the city 1. Asked what kind of routes selected in order to make the shortest total travel (known as the distance between the cities). Since the problem of application has been widely used in many areas, and thus to find the effective algorithms becomes very important.

Genetic algorithm (GA) [2] is a simulation of Darwinian biological evolution by natural selection and genetic mechanism of the biological evolution of the computational model, is a natural evolutionary process by simulating the optimal solution search methods. This paper proposes a hybrid genetic algorithm. It presents two features. One is based on ant algorithm, taking left heuristic information of the previous generations into account in the search process of the optimal solution, and joining the path which generated by this way into new population in genetic algorithm. Another is that genetic algorithm use a local optimization algorithm.

In the remainder of the paper, section II explains ant algorithm. Section III explains local search algorithm and hybrid genetic algorithm. The studies of experiments are performed in section IV. Conclusions and future research are drawn in section V.

II. INTRODUCTION OF ANT ALGORITHM[3]

A. Basic Concept

In order to solve the TSP problem, a group of simple ant agents which called artificial ants are introduced to simulate the natural world really ants. The behavior characteristics of artificial ant reflected in that exist in discrete states, and when move from one state to another using the transfer rules of probability. In two ways, the length of $edge(i, j)$ and the pheromone that store in $edge(i, j)$, which affect the status information of the probability. In general, at time t, the path selection probability of ant k moves from node i to j is

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)^{\alpha}][\eta_{ij}(t)^{\beta}]}{\sum\limits_{j\in A_k}[\tau_{ij}(t)^{\alpha}][\eta_{ij}(t)^{\beta}]} & j\in A_k\\ 0 & other\end{cases} \tag{1}$$

Where $\tau_{ij}(t)$ is the concentration of pheromone on side $arc(i, j)$ at time t, is the communication channel between ants. d_{ij} is the length of the edge (i, j), $\eta_{ij}(t)$ which take $1/d_{ij}$ is a certain heuristic rules. A_k is the set of nodes that ant k allows to select at next step. Parameters α, β represent the relative weights are to adjust the relative importance of side length and pheromone. α is larger, the more ants tend to use the path other ants choose, reflecting the collaboration between ants, β is larger, the selection of probability close to the greedy rule.

According to the constraint in TSP, ants can only access the nodes which have not been to before traveling to complete. Therefore it is necessary to set a taboo table $t[m][n]$. Among them, $t[k][t]$ says the ant k have visited the point before time t. Ants construct a feasible solution of the problem by a full travel around. When m ants have completed a tour, they left pheromone in the respective edge which they through. If ant k has been the $edge(i,j)$ in the tour, the concentration of pheromone that ant k remain on it is

$$\Delta\tau^{k}_{ij} = Q / Z_k \tag{2}$$

Where Q is a constant, Z_K is the total length that ant k walked in this tour loop, corresponding to a feasible solution to the problem. At the end of the first t search cycles, the total concentration of pheromone increment which remain on the $edge(i,j)$ is

$$\Delta\tau_{ij}(\mathrm{t}) = \sum_{\mathrm{k}=1}^{m} \Delta\tau_{ij}^{\mathrm{k}}(t) \tag{3}$$

As the pheromone on the earth will gradually disappear over time, the artificial pheromone evaporation mechanism is also introduced. If the pheromone keep ratio for ρ, the volatile ratio 1-ρ, said the disappearance of pheromone levels, then the pheromone update formula on the $edge(i,j)$ is

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij},\ \rho \in (0,1) \tag{4}$$

B. *Pseudo Code*

Step1. Initialize parameters and set m ants on n cities

Step2. For m ants, select next city with probability $p_{ij}^{k}(t)$, record ant's position and update local pheromone, if m ants finish the complete path, go to Step3, else go to Step2

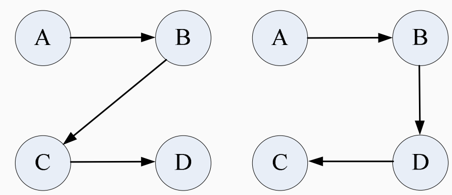
Step3. Record the walked path of every ant and fitness value, update global pheromone

Step4. If running times is less than the specified generation, program goes to Step2, else output the most satisfactory solution

III. Local Search Algorithm and Hybrid Genetic Algorithm

A. *Local Search Algorithm*

In this section it presents a local search algorithm [4], the key point is if after exchange the visit order between adjacent cities, these access paths to the city's total distance is shorter than before the exchange, so this exchange operations is executed. As shown in figure 1, if before local search, the order that through cities A , B, C, D shows as figure (a), after the local search, the order that through cities A, B, C, D shows as figure (b). It is known from the figure that the visited distances of the city after the exchange are shorter than the distances have walked before the exchange, so this exchange execution.



(a) Before the exchange (b) After the exchange

Fig 1. Local search transform figure

B. *Hybrid Genetic Algorithm*

The proposed hybrid genetic algorithm [5], which base on a simple genetic algorithm, use the local search algorithm to speed up the convergence of the algorithm, use ant algorithm to guide the algorithm toward the optimal solution. As the use of the heuristic information of previous generations, it reduces the randomness of search process, improves efficiency of the search population. Use an individual which runs ant algorithm to find heuristic information, and then get the path that the individual visited. Let it to alternative the second-worst individual which was produced by genetic algorithm. The specific processes show in Fig2.

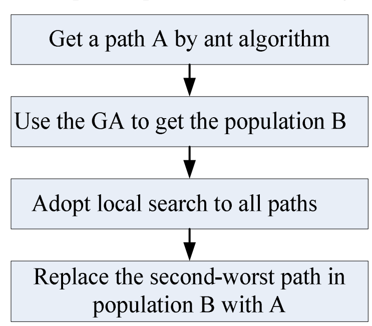


Fig 2. Part flow chart

Specific algorithm is as follows.

Step1. Initialize population and the parameters

Step2. Calculate the fitness of initial population

Step3. Use the ant algorithm to get a heuristic path

Step4. For contemporary population, adopt genetic operation (selection, crossover, mutation and so on) [7], local search, and replace the second-worst path in contemporary population with heuristic path

Step5. Adopt local search to all paths

Step6. Calculation the fitness of all individuals in current population

Step7. If the algorithm moves to the designated generation, then the algorithm is terminated, or turns to Step2

IV. EXPERIMENT STUDY

A. Test Cases

In order to verify the effectiveness of the algorithm, the use of TSPLIB [6] test cases Berlin52, Eil76, lin105, KroA150, KroA200 for experiment, which, Berlin52 has 52 cities, Eil76 have the size of the city for 76, lin105 has 105 cities, KroA150 has 150 cities, and KroA200 has 200 cities.

B. Parameter Setting

In order to compare the algorithms HGA with SGA [8], the general parameters of algorithm is set to as shown in table I. Where HGA is the abbreviation of hybrid genetic algorithm [9], SGA is the abbreviation of simple genetic algorithm.

Table I. PARAMETER SETTINGS

Running generations	1000
Mutation probability	1.0/*CITYNUMS*
Crossover probability	0.90
Population size	150
Ant size	3
α	1
β	5

C. Experimental Results

The test results of test case Berlin52 show in table II, the visited situation of cities show in Fig3.

Table II. TEST RESULTS OF BERLIN52

Algorithm	Optimal solution of Test case	Best solution of experiment	Running times of experiment	Average solution
HGA	7542	7544.366	30	7544.366
SGA	7542	11477.251	30	12380.5

The test results of test case Eil76 show in table III, the visited situation of cities show in Fig4.

Table III. TEST RESULTS OF EIL76

Algorithm	Optimal solution of Test case	Best solution of experiment	Running times of experiment	Average solution
HGA	538	544.369	30	547.3
SGA	538	1138.372	30	1250.2

The test results of test case Lin105 show in table IV, the visited situation of cities show in Fig5.

Table IV. TEST RESULTS OF LIN105

Algorithm	Optimal solution of Test case	Best solution of experiment	Running times of experiment	Average solution
HGA	14379	14382.996	30	14402.4
SGA	14379	52114.253	30	58252.7

The test results of test case KroA150 show in table V, the visited situation of cities show in Fig6.

Table V. TEST RESULTS OF KROA150

Algorithm	Optimal solution of Test case	Best solution of experiment	Running times of experiment	Average solution
HGA	26524	26524.863	30	26789.2
SGA	26524	135759.166	30	144203.3

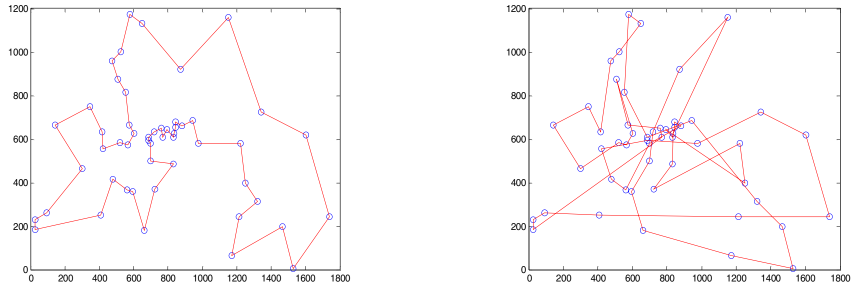
The test results of test case KroA200 show in table VI, the visited situation of cities show in Fig7.

Table VI. TEST RESULTS OF KROA200

Algorithm	Optimal solution of Test case	Best solution of experiment	Running times of experiment	Average solution
HGA	29368	29538.519	30	29934.9
SGA	29368	201805.139	30	212966.2

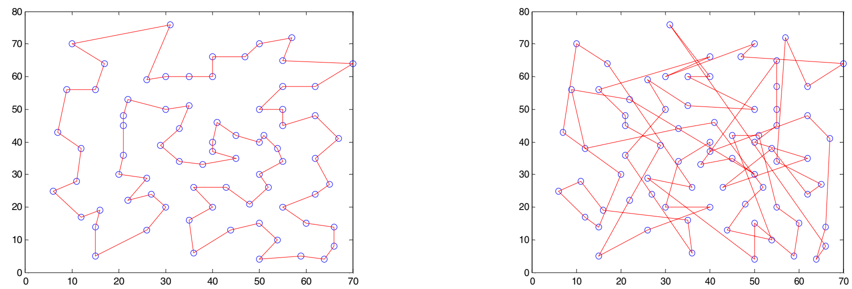
D. Experimental Analysis

As shown from the above, HGA gets better results than SGA in all test cases. So it confirms that local optimal method works well or ant algorithm guides the search towards the optimal solution. Further analysis can find that local optimal method [10] is the main factor for improving the performance of HGA. Ant algorithm accelerates the convergence of HGA to some extent.



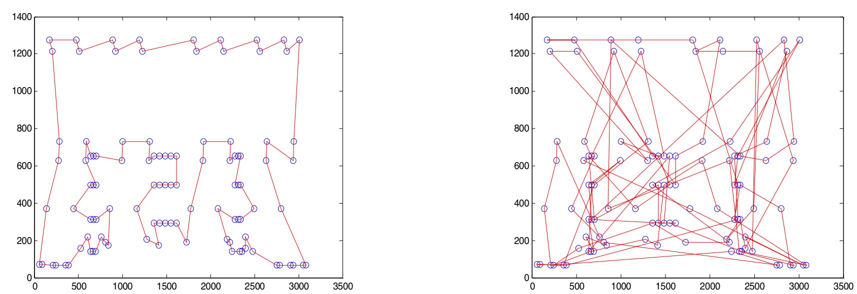
(a) HGA (b) SGA

Fig 3. The visited figure of all cities



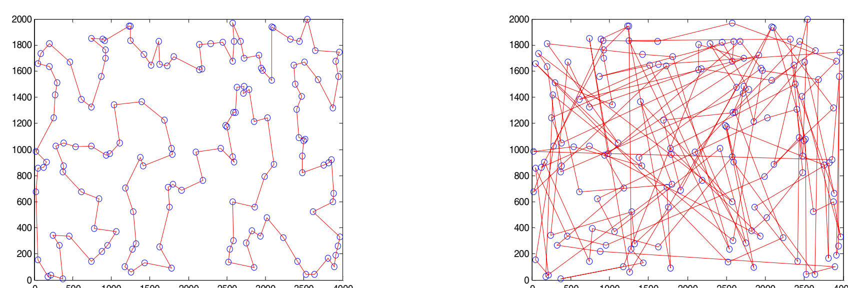
(a) HGA (b) SGA

Fig4. The visited figure of all cities



(a) HGA (b) SGA

Fig5. The visited figure of all cities



(a) HGA (b) SGA

Fig6. The visited figure of all cities

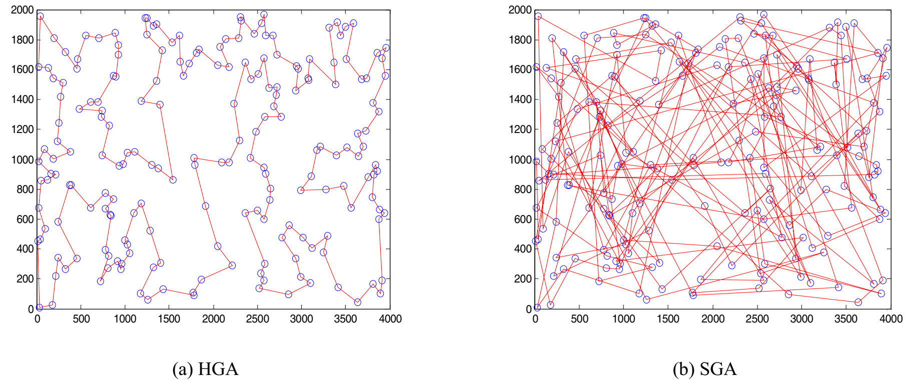


Fig7. The visited figure of all cities

V. CONCLUSION AND FUTURE RESEARCH

This article bases on a simple genetic algorithm. Heuristic information which gets by ant algorithm guides algorithm approaches to the optimal solution. And local search algorithm helps to accelerate the convergence of hybrid genetic algorithm. By contrast with the simple genetic algorithm, hybrid genetic algorithm obtains better results, but there are still many deficiencies to be improved. Such as, when compare it with other good algorithms, the number of test city is insufficient and some test cases don't find the optimal solution. This algorithm is time-consuming due to add local search operations. For these reasons, it is necessary to design some faster optimal operations for time-consuming problem. Maybe we can use density information between cities for TSP problem, and these will be my next research.

REFERENCES

[1] Sushil J. Louis. Genetic Algorithm for TSP [OL]

[2] Goldberg D E, Lingle R J, Alleles, Loci, and the Traveling Salesman Problem, In:Proc, of first Int.Conf.on Genetic Algorithms and Theory, Applications, Lawrence Erlbaum Associates, 1985, 154-159

[3] Drigo M, Maniezzo V and Colorni A. The ant system: optimization by a colony of cooperating agents〔J〕. IEEE Trans. Syst, Man, Cybern. B, 1996. 26 (2) : 29～41

[4] Lin S, Kernighan B W. Effective Heuristic Algorithm for the Traveling-Salesman Problem [J]. Operation Research (S1526-5463), 1973, 21(2): 498-516.

[5].Katayama K AND Sakamoto H. The Efficiency of Hybrid Mutation Genetic Algorithm for the Travelling Salesman Problem. Mathematical and Computer Modelling [J], 2000, 31:197-203

[6].TSPLIB Http://www.Iwr.uni_heidelberg.de， May 22,2007

[7] Tao G, Michalewicz Z. Inver-over Operator for the TSP [C].Proceedings of the 5 Parallel Problem Solving form Nature, Lecture Notes in Computer Science,1998,1498:803-812

[8] Yang Ping, Zheng Jinhua. The comparison and research of genetic selection operator [J].Computer engineering and Applications, 2007, 43(15):59-65.

[9] Baraglia R, Hidalgo J I, Perego R.A hybrid heuristic for the traveling salesman problem [J].IEEE Transactions on Evolutionary Computation,2001,5:613-622

[10] J.Stender: Parallel Genetic Algorithms: Theory and Application [M], IOS Press, 1993